

TM-S : 서비스를 위한 Topic Maps 기반의 온톨로지 및 질의 언어 설계*

유정연⁰ 이규철
충남대학교 컴퓨터공학과
{jyou⁰, kclee}@ce.cnu.ac.kr

Define the Ontology and Query Language Based on Topic Maps for Service

Jeong-Youn Yu⁰, Kyu-Chul Lee
Dept. of Computer Engineering, Chungnam National University

요 약

대표적인 시맨틱 웹 서비스 발견 기술은 OWL-S와 MIT의 Process Handbook이 있다. 그러나, OWL-S는 개발 초기 단계이기 때문에, 아직 효과적인 웹 서비스 발견을 제공하기에는 몇 가지 제약 조건을 가지고 있다. 예를 들어, 정보 전송을 위한 제약 조건과 실행에 따른 상태 변환 정보를 정의하고 있지 않다. 또한, 사용자가 원하는 프로세스들의 시맨틱 정보들을 정의하고 있지 않다. 반면, MIT Process Handbook은 OWL-S와 같이 서비스 모델에 대한 상세한 정보들을 정의하고 있지 않아, 서비스 작성에 필요한 서비스들을 찾기가 어렵다. 그러므로, 본 논문에서는 Topic Maps 기반의 TM-S(Topic Maps for Service)를 제안하였다.

1. 서 론

대표적인 시맨틱 웹 서비스 발견 기술은 OWL-S(OWL-based Web Service Ontology)와 MIT(Massachusetts Institute of Technology)의 Process Handbook이 있다 [1][2]. OWL-S는 서비스 발견에 사용되어지는 서비스의 속성이나 연산들을 기술하는 서비스 온톨로지이다. MIT의 Process Handbook은 서비스 모델에서 사용자에게 의미있는 서비스의 역할과 관계 정보들을 정의한다. 이렇게 정의된 서비스 모델들은 프로세스 온톨로지에 인덱싱함으로써 검색을 유용하게 한다. 그러나, OWL-S는 개발 초기 단계이기 때문에, 아직 효과적인 웹 서비스 발견을 제공하기에는 몇 가지 제약 조건을 가지고 있다. 예를 들어, 정보 전송을 위한 제약 조건과 실행에 따른 상태 변환 정보를 정의하고 있지 않다. 또한, 사용자가 원하는 프로세스들의 시맨틱 정보들을 정의하고 있지 않다. 반면, MIT Process Handbook은 OWL-S와 같이 서비스 모델에 대한 상세한 정보들을 정의하고 있지 않아, 서비스 작성에 필요한 서비스들을 찾기가 어렵다. 그러므로, 본 논문에서는 Topic Maps 기반의 TM-S(Topic Maps for Service)를 제안하였다. TM-S 기술은 OWL-S와 MIT Process Handbook의 장점을 지원할 뿐만 아니라 단점을 보완하는 온톨로지를 정의한다. TM-S의 기반이 되는 Topic Maps은 서비스 발견을 유용하게 하기 위해 서비스를 분류하고 체계화하며, 상호 교환하기 위해 XML 기반의 XML Topic Maps(XTM)을 정의하고 있다[4]. 이것은 ISO 국제 표준으로 승인된 상태이다.

* 본 연구는 한국과학재단 지정 충남대학교 소프트웨어연구센터(SOREC)와 정보통신부의 대학 IT연구센터(ITRC) 지원을 받아 수행되었습니다

또한 TM-S의 기능을 질의 할 수 있는 TMS-QL(Query language for TM-S)를 정의하였다.

2. Topic Maps for Service(TM-S)

기능적인 웹 서비스 발견은 요구하는 제약 조건을 만족하는 특정한 웹 서비스의 검색이 필요하다. 이를 위해서는 서비스의 일반적인 특정 뿐만 아니라, 서비스를 구성하는 하위 프로세스나 프로세스의 흐름 정보와 같은 좀 더 깊이 있는 정보들을 제공하는 온톨로지의 구축이 절실히 요구된다. 그리고, 서비스의 탐색을 쉽게 하기 위해, 사용자의 관점에 따라 서비스를 나누고, 서비스들간의 관련 정보들을 정의하는 것이 필요하다. 이와 같은 작업은 Topic Maps 기술을 이용하여 쉽게 표현할 수 있다.

Topic Maps은 정보 자원들을 분류하고 체계화하여 지식으로 변환하는 온톨로지 기술이다. 이 Topic Maps은 3가지의 기본 개념을 이용하여 정보 자원들의 구조와 관련 정보를 정의한다. 이 개념은 topic, association, occurrence(TAO)이다. Topic은 컴퓨터에서 실 세계 객체 정보를 구체화하는 자원이다. 이 Topic은 이름과 occurrence를 가진다. 이 occurrence는 주어진 객체 정보를 명확히 하기 위해 관련 정보 자원들을 정의한다. 마지막으로, topic은 association이라는 관계 정보 안에서 구성 요소로서 수행할 수 있다. 또한, 이렇게 정의된 자원 정보들은 특정한 범위나 상황에서 유효화 되어지도록 정의하고 있다. 현재는 웹 기반으로 topic map정보를 상호

이 논문에서는 Topic Maps for Service(TM-S)라고 불리는 서비스 온톨로지와 질의 언어를 정의하였다. 그림 1은 TM-S 온톨로지의 범위를 보여준다. TM-S는 OWL-S와 MIT Process Handbook 온톨로지를 모두 지원하면서, OWL-S와 Process Handbook의 약정을 보완한다. 즉, 서비스 이름, 관계자, 입력/출력 파라미터등과 같이 OWL-S의 간결하고 능률적인 모델링을 표현한다. 또한, 매칭 능력을 쉽게 하기 위해, MIT Process Handbook에서 정의한 서비스의 분류 정보를 구성한다. 이 분류 정보는 서로 다른 서비스나 프로세스의 관점에 따라 다르게 기술 되어질 수 있다. 마지막으로, 본 연구는 OWL-S에서 아직 완전히 정의되지 않은 서비스의 기능(입력/출력의 제약 조건, 결과/선행조건에 대한 룰)들을 정의하는 것을 목표로 하고 있다.

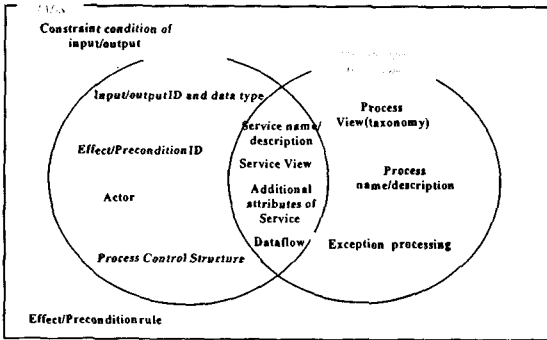


그림 1. TM-S 온톨로지의 범위

3. TM-S 모델링

TM-S를 모델링하기 위해, OWL-S 명세에서 서비스 찾기 위해 사용되어 질 수 있는 프로파일과 모델을 위한 온톨로지를 이용한다. 또한, MIT Process Handbook의 주요 개념인, 서비스 모델을 인식하는 프로세스 온톨로지를 정의한다. 그러나, 본 연구에서는 아직 입력/출력의 제약 조건 정의, 결과/선행 조건 룰, 예외 처리에 해당되는 정의는 포함하지 않고 있다. 현재, 룰 기반의 언어를 이용하여 이러한 문제를 풀기 위한 연구를 진행 중에 있다.

또한, TM-S는 OWL-S와 MIT Process Handbook에서 제공되는 기능을 지원하면서, 추가로 제한된 상황에 따라 서비스/프로세스에 대한 다양한 이름을 할당한다. 예를 들어, TM-S는 국가 언어, 표준 언어, 축약어 등에 따라 서비스/프로세스의 이름을 정의할 수 있다. 마지막으로, TM-S의 구조를 유효하기 위해 OSL(Ontopia Schema Language)를 이용하여 표현하였다[3]. TM-S 모델링에 대한 정의는 다음 논문에서 정의되어 있으므로 생략하도록 하겠다[5].

교환하기 위해 XML 기반의 Topic Maps을 정의하였다.

4. TM-S를 위한 질의 언어

본 연구는 TM-S를 위해 2 가지의 질의 능력을 제공한다. 즉, 브라우징 질의와 상세 질의이다. 브라우징 질의는 사용자가 관심 있어 하는 여러 개의 뷰나 온톨로지를 단순히 선택함으로써 서비스나 프로세스를 찾는 기술이다. 만일 사용자가 'sell', 'buy', 'pay', 'identify' 등과 같은 서로 다르게 분류되어 있는 온톨로지들을 선택한다면, 모든 온톨로지에 정의된 서비스 정보들에 대한 논리적인 OR 연산을 처리한다. 그렇지 않을 경우는, 논리적인 AND 처리를 수행한다. 상세 질의는 TM-S 모델을 기반으로 향상된 질의 능력을 제공하는 기술이다. 이번 절에서는 이러한 상세 질의를 지원하는 TMS-QL(Query Language for TM-S) 질의 언어를 설계하였다.

4.1 TMS-QL을 위한 스키마

표 1에서와 같이, 본 논문에서는 서비스나 프로세스의 이름에서부터 간단한 프로세스의 구조를 질의할 수 있는 TMS-QL을 위한 스키마를 정의하였다. 이 스키마는 6개의 클래스로 구성되어 있다. 즉, Service, ServiceProfile, Actor, ProcessModel, Process, ServiceOntology 이다. 이 클래스들은 TM-S 온톨로지에 정의 되어있는 애트리뷰트들에 의해 특성화 되어진다. 또한, 클래스 ServiceOntology를 중심으로 서로 서로 연관 관계를 형성한다. 그림 2는 TMS-QL을 위한 질의 스키마를 UML로 표현한 것이다.

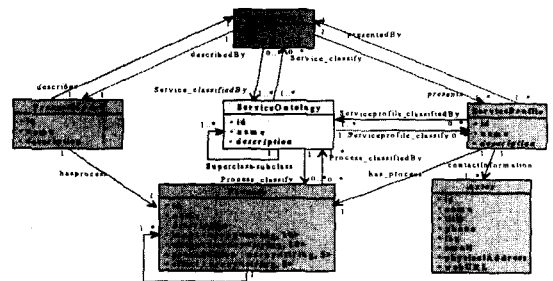


그림 2. TMS-QL의 UML 클래스 다이어그램

4.2 TMS-QL을 위한 Algebra

이번 절에서는 TM-S 온톨로지를 조작하기 위해 필요한 algebra 연산을 기술한다. 이 연산자들은 그림 2에서 보여진 TMS-QL을 따르는 객체들을 고려한다.

$Project(\pi) : \pi [C; A]$ 는 클래스 C에 있는 애트리뷰트 A에 대한 객체 컴포넌트들을 정의한다. 예를 들어,

표 1. TMS-QL의 algebra 연산들

Operator	Argument	Query Example
Project (π)	Object, AttributeName	ServiceProfile.name, ServiceProfile.*
Scoped Project (π')	Object, AttributeName, ScopeType	Service.name["korean"]
Select (σ)	Object, AttributeName, CompareString, operator	Actor.name like "CNU", ServiceProfile.name contains('computer AND 'book')
Scoped Select (σ')	Object, AttributeName, ScopeType, CompareString, operator	Actor.name["korean"] like "%출판사%"
Association (Ω)	FromObject, ToObject, AttributeName	Service->>ServiceProfile.name
Scoped Association (Ω')	FromObject, ToObject, AttributeName, ScopeType	Service->> ["korean"] ServiceProfile.name
Scoped Ontology (λ)	ScopeType, attributeName, CompareString, operator	ServiceOntology-["sell what", name like 'sell via Store']
Method	Object, MethodName, Depth	Service.subProcess(3)
Union (\cup)	Object1, Object2	Result1 or Result 2
Intersection (\cap)	Object1, Object2	Result1 and Result 2
Difference ($-$)	Object1, Object2	Result1 - Result2

ScopedProject(π') : 이것은 특정한 상황에서 유효한 애트리뷰트들에 대해 동작하는 Project 연산이다. 즉, π' [C; A; S]은 π [C; A]의 결과에서 상황 S를 만족하는 객체들의 집합을 넘겨준다. 본 연구에서는 TMS-QL 신택스에서 상황 정보를 기술하기 위해 “[]” 연산자를 이용한다. 표 2에 있는 ScopedProject 연산에 대한 질의 예는 name 애트리뷰트를 제한한것이다. 즉, 서비스의 이름이 한국어로 쓰여진 서비스들을 찾아달라고 하는 질의이다.

Select(σ) : Select 연산은 선택 조건을 만족하는 객체들의 집합을 선택하기 위해 사용된다. σ [C; A; st; f]에서 st는 애트리뷰트도메인의 상수 값이며, f는(‘=’, ‘like’, ‘contains’) 연산자 중 하나이다. 만일, ‘like’ 연산자가 사용된다면, 글자 스트링의 일부분만을 비교한다. 그렇지 않을 경우, ‘contains’ 연산자를 사용한다면, 정의된 단어를 포함하는 텍스트들을 검색한다

ScopedSelect(σ') : 이것은 특정한 상황에서 선택 조건을 제공하는 연산으로서 Select 연산을 확장한 것이다. 표 2의 예에서, ScopedSelect 연산은 관계자의 이름이 한국어로 ‘출판사’인 객체들을 넘겨준다.

Association(Ω) : 이것은 2개의 클래스에 정의된 객체들 간의 복잡한 관계들을 구성하는 컴포넌트들의 검색을 유용하게 하는 연산이다. 이 Association 연산은 Ω [Cf; Ct; A]로 구성되어 있으며, Cf는 처음 클래스, Ct는 목표 클래스이며, A는 목표 클래스의 애트리뷰트를 정의한다. 즉, Cf 클래스의 객체를 참조하는 Ct 클래스 객체의 컴포넌트들을 넘겨주게 된다.

ScopedAssociation(Ω') : ScopedAssociation과 ScopedProject 연산은 주로 특정 상황에 유효한 객체 컴포넌트들을 넘겨주기위해 사용된다. ScopedProject 연산은 하나의 관계에 있는 객체 컴포넌트들에 대해 사용되며, 반면에 ScopedAssociation 연산은 서로 다른 타입의 객체들이 서로 관계를 가질 때 사용된다

π [ServiceProfile; name]은 ServiceProfile의 이름을 나타내는 객체들의 집합을 넘겨준다.

ScopedOntology(λ) : 이 연산은 Service Ontology의 특정 상황에 유효한 객체들을 다룬다. 표 2의 예에서 ‘sell what’ 상황에서 ‘sell what store’ 온톨로지 안에 있는 서비스들을 찾는 것이다.

Method : 본 논문에서는 서비스를 구성하는 프로세스들을 찾기 위해 4가지의 메소드를 정의하였다. 이 aptem는 트리 형태로 구성되어 있는 온톨로지 구조에 동일하게 적용될 수 있다.

subProcess(int depth) : 깊이 값을 정의하여 하위 프로세스들을 넘겨준다.

superProcess(int depth) : 깊이 값을 정의하여 하위 프로세스들을 넘겨준다.

rootProcess() : 최 상위 프로세스를 넘겨준다.

leafProcess() : 최 하위 프로세스를 넘겨준다.

Union(\cup), Intersection(\cap), Difference($-$) : 3개의 연산은 같은 타입을 가지는 정보를 받아 집합 연산을 수행한다.

5. 결론

OWL-S와 MIT Process Handbook 온톨로지의 연합은 웹 서비스 발견의 능력을 향상시킬 수 있다. 본 연구는 이러한 작업의 수행을 도울 수 있는 TM-S 온톨로지를 표현함으로써 이러한 가능성을 보여주고 있다. 또한, TM-S 온톨로지의 능력을 처리 할 수 있는 질의 언어도 기술하였다. 이것은 웹 서비스 발견을 위한 현재의 기술을 향상시키는 새로운 방향을 제공한다고 할 수 있다. 현재, 다음과 같은 연구를 진행 중에 있다.

- 사용자가 기대하는 서비스를 정확하기 매치하기 위한 지능적인 matchmaking 기술 설계
- 복잡한 질의가 가능한 질의 언어의 향상
- 입력/출력 제약과 선행 조건/결과 등을 정의 할 수 있는 룰 기반의 정의 및 구현

6. 참고문헌

[1] A. Ankolekar, M. Burstein, J. Hobbs, O. Lassila, D. Martin, S. McIlraith, S. Narayanan, M. Paolucci, T. Payne, K. Sycara, and H. Zeng : OWL-S : Semantic Markup for Web Services.

[2] Bernstein, A., Klein, M. : Towards High-Precision Service Retrieval. Proceedings of The International Semantic Web Convergence(2002) 84-101

[3] Ontopia. : The Ontopia Schema Language : Tutorial, Available at <http://www.ontopia.net/omnigator/docs/schema/tutorial.html>.

[4] Steve P. : The TAO of Topic Maps : Finding the Way in the Age of Information. Proceedings of XML Europe 2000, GCA(2000).

[5] 황윤영, 유정연, 유소연, 이규철, 시맨틱 웹 서비스를 위한 Topic Maps 기반의 온톨로지 언어, u-korea를 위한 전자거래 종합학술대회 논문집, 한국전자거래학회, 여의도 전경련회관, pp.191-196 (2003. 9. 4)